Contents lists available at ScienceDirect



ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs



# Deep learning for conifer/deciduous classification of airborne LiDAR 3D point clouds representing individual trees



Hamid Hamraz<sup>a,\*</sup>, Nathan B. Jacobs<sup>a</sup>, Marco A. Contreras<sup>b,c</sup>, Chase H. Clark<sup>b</sup>

<sup>a</sup> Department of Computer Science, University of Kentucky, Lexington, KY 40506, USA

<sup>b</sup> Department of Forestry, University of Kentucky, Lexington, KY 40506, USA

<sup>c</sup> Instituto de Bosques y Sociedad, Facultad de Ciencias Forestales Y Recursos Naturales, Universidad Austral de Chile, Campus Isla Teja, Valdivia, Chile

#### ARTICLE INFO

Keywords: Remote sensing Convolutional neural network Representation engineering Unbalanced training data Mislabel correction

## ABSTRACT

The purpose of this study was to investigate the use of deep learning for coniferous/deciduous classification of individual trees segmented from airborne LiDAR data. To enable processing by a deep convolutional neural network (CNN), we designed two discrete representations using leaf-off and leaf-on LiDAR data: a digital surface model with four channels (DSM  $\times$  4) and a set of four 2D views (4  $\times$  2D). A training dataset of tree crowns was generated via segmentation of tree crowns, followed by co-registration with field data. Potential mislabels due to GPS error or tree leaning were corrected using a statistical ensemble filtering procedure. Because the training data was heavily unbalanced ( $\sim$ 8% conifers), we trained an ensemble of CNNs on random balanced sub-samples. Benchmarked against multiple traditional shallow learning methods using manually designed features, the CNNs improved accuracies up to 14%. The  $4 \times 2D$  representation yielded similar classification accuracies to the DSM  $\times$  4 representation (~82% coniferous and ~90% deciduous) while converging faster. Further experimentation showed that early/late fusion of the channels in the representations did not affect the accuracies in a significant way. The data augmentation that was used for the CNN training improved the classification accuracies, but more real training instances (especially coniferous) likely results in much stronger improvements. Leaf-off LiDAR data were the primary source of useful information, which is likely due to the perennial nature of coniferous foliage. LiDAR intensity values also proved to be useful, but normalization yielded no significant improvement. As we observed, large training data may compensate for the lack of a subset of important domain data. Lastly, the classification accuracies of overstory trees (~90%) were more balanced than those of understory trees (~90% deciduous and ~65% coniferous), which is likely due to the incomplete capture of understory tree crowns via airborne LiDAR. In domains like remote sensing and biomedical imaging, where the data contain a large amount of information and are not friendly to human visual system, human-designed features may become suboptimal. As exemplified by this study, automatic, objective derivation of optimal features via deep learning can improve prediction tasks in such domains.

#### 1. Introduction

Remote sensing technologies have long been a means to facilitate data acquisition over large forested areas (Franklin, 2001). For instance, aerial images have been used to map forests and monitor their growth and regeneration (Gougeon, 1995; Pitkänen, 2001; Quackenbush et al., 2000). However, 2D images, as snapshots of the 3D world, lack depth information and are insufficient for more detailed tasks such as derivation of vertical canopy structure, biomass quantification, or segmentation of individual trees. Airborne light detection and ranging (LiDAR) directly measures depth and can capture multiple

returns per pulse, thereby representing the forested landscapes in the form of 3D point clouds (Ackermann, 1999; Hyyppä et al., 2012; Maltamo et al., 2014). These point clouds can be processed to segment individual trees (Amiri et al., 2016; Jing et al., 2012; Kwak et al., 2007; Paris et al., 2016; Popescu and Zhao, 2008; Sačkov et al., 2017; Véga et al., 2014; Wang et al., 2008), which enable deriving individual tree attributes such as height, crown width, and allometric relationships, as well as predicting individual tree parameters such as type, species, status (live or dead), or diameter at breast height (DBH) (Duncanson et al., 2015; Vauhkonen et al., 2010; Yu et al., 2011).

Tree species information is important for appropriate stem biomass

\* Corresponding author.

https://doi.org/10.1016/j.isprsjprs.2019.10.011

Received 24 November 2018; Received in revised form 17 October 2019; Accepted 18 October 2019

E-mail addresses: Hamraz@cs.uky.edu (H. Hamraz), jacobs@cs.uky.edu (N.B. Jacobs), marco.contreras@uach.cl (M.A. Contreras), chase.clark2@uky.edu (C.H. Clark).

<sup>0924-2716/ © 2019</sup> International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

and dimension estimation, hence is essential for optimal management decisions (Holmgren and Persson, 2004; Holopainen and Talvitie, 2007; Korpela et al., 2010). Several studies have used segmented point clouds representing individual trees to predict tree type (coniferous or deciduous) or species using machine learning methods (Blomley et al., 2017; Cao et al., 2016; Harikumar et al., 2017; Holmgren and Persson, 2004; Kim et al., 2011; Lindberg et al., 2014; Ørka et al., 2009; Reitberger et al., 2008). In these studies, researchers derived a set of features related to crown geometry and foliage density/pattern/texture from the LiDAR data and input the features into different classification methods such as linear discriminant analysis (LDA), k-nearest neighbors (KNN), random forest, and support vector machines (SVMs). A few studies have presented automated or semi-automated approaches for identifying useful features for the task of tree species classification (Bruggisser et al., 2017; Li et al., 2013; Lin and Hyyppä, 2016). Previous work using traditional learning methods has required that the set of candidate features be assembled by an expert, with the intention of removing redundant and less useful information from the raw data. However, given the large amount of information contained in LiDAR point clouds and their unfriendliness to human eyes, the expert-designed features may be suboptimal.

Deep neural network learning methods, on the other hand, can directly map the raw input data to the target prediction (LeCun et al., 2015; Schmidhuber, 2015). These methods pass the input through multiple layers where, conceptually, the initial layers extract the useful low to mid-level features and the next layers map the extracted features to the target prediction. Designing a deep network architecture and tuning the training hyper-parameters require some level of expertise. However, the end result is typically not sensitive to these preliminary steps as long as the choices fall within a reasonable range (Bergstra and Bengio, 2012). In fact, training a deep network including feature extraction and mapping them to the target predictions runs as one unified, end-to-end optimization process; the process fits the network parameters into any reasonably chosen architecture/hyper-parameters such that the global prediction task functions optimally. In contrast to traditional learning methods, this optimization objectively derives features according to the target prediction task and minimizes subjectivity and bias in the features extracted.

A large body of research has been devoted to a variety of deep learning classification or segmentation tasks using 2D images as the raw input data (Girshick et al., 2014; Krizhevsky et al., 2012). However, 3D data have been considered less often until recently, which is due to more costly acquisition/processing and their less intuitive and less conventional representational formats. Unlike 2D images that can readily be processed by a convolutional neural network (CNN) architecture, 3D data require designing the appropriate representation to make them usable for deep learning methods (Qi et al., 2017a; Qi et al., 2016). A number of studies have binned 3D data into voxel spaces to create representations that can be input to and processed by a 3D CNN (Dou et al., 2016; Maturana and Scherer, 2015; Wu et al., 2015). Although voxel spaces are perhaps the most comprehensive discrete representations that preserve the raw 3D structure, they are computationally expensive to process, more prone to overfitting, and therefore prohibitive for use with larger datasets. Taking advantage of sparsity and non-uniformity of 3D data, a few recent studies have proposed alternative approaches that hierarchically index the 3D space according to the regional data density, hence lowering the computational cost (Qi et al., 2017b; Riegler et al., 2017). These approaches take the raw point clouds as the input and do not require any representation design, but they use modified versions of the convolution and pooling operations to conform to the indexing structure. Other studies have created 2.5D digital surface models (DSMs) (Mizoguchi et al., 2017; Roth et al., 2016; Socher et al., 2012) or multiple 2D views (Farfade et al., 2015; Su et al., 2015) from the 3D data. If the 3D imaging/sensing technology is able of capturing the internal structure of the measured objects, conversion to DSM or 2D views may forego this internal structure.

However, depending on the application, DSMs and/or multiple 2D views can provide as much useful information as a full 3D representation while being less prone to overfitting and incurring less computational cost (Kalogerakis et al., 2017; Su et al., 2015).

A few recent studies used deep learning methods to classify species of individual trees from very high-resolution ground-based LiDAR point clouds. Guan et al. (2015) segmented individual trees from mobile LiDAR point clouds in an urban area, developed a waveform representation to model the geometry of the trees, and used deep learning to convert the waveform representation to high-level features. These features were then input to an SVM classifier to perform tree species classification. Mizoguchi et al. (2017) also segmented individual trees from terrestrial LiDAR point clouds, derived DSM patches representing the tree bark texture from the clouds, and fed this information into a CNN to perform classification between two species.

In this paper, we segment individual trees from airborne LiDAR data representing a natural dense forest, prepare segmented crowns for input to a CNN, and perform different shallow and deep learning experiments on tree type classification. The main contributions of this work are to: (i) design two discrete representations with minimal loss of information for the 3D crown cloud captured by airborne LiDAR, (ii) benchmark performance of deep CNN learning against shallow learning, and (iii) investigate the effects of different design decisions with respect to training data preparation, CNN design, training data composition, and inclusion of domain-specific data on the classification accuracy.

## 2. Materials and methods

## 2.1. Study site, LiDAR campaign, and field survey

The study site is the University of Kentucky's Robinson Forest (RF, Lat. 37.4611, Long. -83.1555). RF is in the rugged eastern section of the Cumberland Plateau region of southeastern Kentucky in Breathitt. Perry, and Knott counties. RF features a variable, dissected topography with moderately steep slopes, which range from 10% to over 100% and face predominately northwest to southeast. Elevation ranges from 252 to 503 m above sea level (Carpenter and Rumsey, 1976). Having been extensively logged in the 1920's, RF is considered a second growth forest ranging from 80 to 100 years old, and it is now protected from commercial logging and mining activities (Department of Forestry, 2007). RF currently extends over an aggregate area of 7,440 ha and includes about 2.5 million ( ± 5.6%) trees (330 stems per ha) (Hamraz et al., 2017b). The average canopy cover is about 93% with small openings scattered throughout. Most areas exceed 97% canopy cover, but recently harvested areas have an average cover as low as 63%. RF features a diverse, contiguous, mixed mesophytic vegetation made up of various deciduous tree species with northern red oak (Quercus rubra), white oak (Quercus alba), yellow-poplar (Liriodendron tulipifera), American beech (Fagus grandifolia), and sugar maple (Acer saccharum) as overstory species. Deciduous understory species include eastern redbud (Cercis canadensis), flowering dogwood (Cornus florida), spicebush (Lindera benzoin), pawpaw (Asimina triloba), umbrella magnolia (Magnolia tripetala), and bigleaf magnolia (Magnolia macrophylla) (Carpenter and Rumsey, 1976; Overstreet, 1984). A small number of conifer species also exists throughout the forest including eastern hemlock (Tsuga canadensis), which can occur in clusters near streams, and different species of Pine (Pinus sp).

The LiDAR data are a combination of two separate datasets collected with the Leica ALS60 LiDAR system (Leica Geosystems). For both datasets, the system was set at 200 kHz pulse repetition rate and 40° field of view, and was flown with an average speed of 105 knots (194.46 Km/h) over strips with 50% overlap. One dataset was low density ( $\sim$ 2pt/m<sup>2</sup>), collected in a day in the spring of 2013 during the leaf-off season (average altitude of 3,096 m above the ground) for the purpose of acquiring terrain information as a part of a state-wide elevation data acquiring program by the Kentucky Division of Geographic

Information. The second dataset was high density ( $\sim$ 50pt/m<sup>2</sup>), collected in three consecutive days in the summer of 2013 during the leafon season (average altitude of 214 m above the ground). Up to three returns per pulse for the leaf-off and up to four returns per pulse for the leaf-on collections were captured, and only 90–95% of the middle portion of the flight strips was used to create the datasets. Both datasets were processed by the vendor using TerraScan software (Terrasolid Ltd., 2012)to classify the LiDAR points into ground and non-ground. The ground points were then used to create a 1-meter resolution digital elevation model (DEM) using nearest-neighbors average method to fill the gaps.

Throughout RF, 271 regularly distributed (grid-wise every 384 m) circular plots of 0.04 ha, centers of which were georeferenced with 5 m accuracy, were field surveyed during the fall of 2013 and spring of 2014. Within each plot, DBH (cm), tree height (m), species, crown class (dominant, co-dominant, intermediate, overtopped), tree status (live, dead), and stem class (single, multiple) were recorded for all trees with DBH greater than 12.5 cm. In addition, horizontal distance and azimuth from plot center to the face of each tree at breast height were collected to create a stem map. Excluding trees below 4 m in height, a total of 3987 trees were surveyed of which 7.27% were conifers (Table 1).

## 2.2. Data preparation

## 2.2.1. LiDAR intensity normalization

The LiDAR intensity value that is recorded for each return is dependent on various factors, many of which are unrelated to the vegetation texture (Gatziolis, 2011; Kashani et al., 2015). The distance a LiDAR pulse travels (referred to as range), the angle at which the pulse is scanned, and the LiDAR return number are among the controllable factors affecting intensity, while different atmospheric factors are difficult to track. Assuming constant atmospheric conditions for the short periods of collections (one day for the leaf-off and three consecutive days for the leaf-on), we used a data-driven approach to normalize the intensity values. We binned the entire forest dataset to a horizontal grid with a cell width of 10 m and randomly sampled one leaf-off and one leaf-on vegetation point per grid cell. We then grouped the leaf-off and the leaf-on samples by the return number, yielding three leaf-off and four leaf-on datasets. For each of the seven datasets, we built a regression model that predicted intensity based on range and scan angle. For the leaf-on datasets, the effect of range and angle was significant: the natural logarithm of range had a negative correlation with intensity (P < .0001), and the cosine of angle has a positive correlation (P < .0001) with intensity. However, we did not observe any significant correlations between range/angle and intensity for the leaf-off datasets. This observation is likely due to the higher flight altitude (longer range), resulting in very low recorded intensity values with small variations such that these correlations faded away. For each of the

	Conifer	Percent in Conifers	Deciduous	Percent in Deciduous	Total	Percent in Total
Dominant	10	3.45%	120	3.46%	130	3.26%
Co-Dominant	39	13.45%	919	24.86%	958	24.03%
Intermediate	78	26.90%	1409	38.12%	1487	37.30%
Overtopped	143	49.3%	1012	27.38%	1155	28.97%
Dead	20	6.90%	236	6.39%	256	6.42%
All	290	100.0%	3697	100.0%	3987	100.0%
Percent of Total	7.27%		92.73%		100.0%	
Species Count	6		37			
Shannon Diversity Index	0.605		2.673			

four leaf-on datasets, we removed the effects of range and scan angle by residualization (Allen, 1997), i.e., we replaced the intensity value of each LiDAR point by the residual (observed minus predicted) value for the point. We then scaled the residualized intensities back to an eightbit format to minimize the effect of return number across the datasets.

## 2.2.2. Individual tree segmentation and registration with field data

We included the points within a 10 m buffer around the LiDAR point clouds corresponding to the 271 field-surveyed plots to capture the complete crowns of border trees. Using the DEM, we calculated the height above ground for the LiDAR points and excluded the points below 3 m (ground level vegetation). We then vertically stratified the point clouds into multiple canopy layers by analyzing the vertical distributions of the LiDAR points within overlapping locales (Hamraz et al., 2017c). We excluded the canopy layers with densities less than 3pt/m<sup>2</sup> from further analysis because tree segmentation at such low densities becomes inaccurate (Evans et al., 2009; Hamraz et al., 2017a). We then segmented each of the canopy layers independently using the method we designed for complex vegetation structures (Hamraz et al., 2016). This method identifies crown boundaries around the global maximum of the canopy layer and clusters the points encompassed by the convex hull of boundary points to complete the segmentation for the tallest tree. This process is repeated until all the points are clustered. Clusters representing crowns less than 1.5 m in average width are finally removed as noise.

After segmentation, we re-calculated the height above ground of the points representing each individual crown according to the median value of DEM beneath the crown to prevent deformation of crown shapes due to DEM variability. To register the segmented crowns with the field data, we assigned a score to each pair of segmented crown and field-measured stem locations. The location of each segmented crown was taken from the crown apex. Scores were assigned based on the difference in tree height and the leaning angle from nadir between the crown apex and the stem location. If the height difference was less than 10% and the leaning angle was less than 5°, a score of 100 was assigned. If the height difference and leaning angle were less than 20% and 10° respectively, a score of 70 was assigned. If the height difference and leaning angle were less than 30% and 15°, a score of 40 was assigned. We then selected the set of pairs with the maximum total score where each crown or stem location appears not more than once using the Hungarian assignment algorithm and regarded the set as the co-registered tree pairs (Hamraz et al., 2016; Kuhn, 1955). Excluding dead trees, a total of 2528 co-registered trees was gleaned, of which 124 (4.90%) were conifers and 2404 (95.10%) were deciduous. Smaller understory trees, especially those represented by very low point densities, were automatically excluded through the segmentation and registration process.

## 2.2.3. Discretization of segmented point clouds

We converted the point cloud of each tree crown to two different representational formats: (1) a DSM with four channels (DSM  $\times$  4), and (2) a set of four single-channel 2D images (4  $\times$  2D). To create the  $DSM \times 4$  format, we binned the point cloud to a horizontal grid of  $128 \times 128$  pixels of width 12.5 cm such that the apex of the segmented crown would fall in the center pixel (Fig. 1). We then recorded the four channel values for each pixel, which included the elevation above ground of the highest leaf-on point, the normalized intensity of the highest leaf-on point, the elevation above ground of the highest leaf-off point, and the intensity for the highest leaf-off point. We chose the small pixel width of 12.5 cm for creating the DSM image to minimize the information loss because of falling multiple LiDAR points in a pixel. The resulting DSM structure captures a square of  $16 \times 16$  m in the real world, which is large enough to encompass an entire tree crown in almost all cases given that tree crowns are often relatively narrow in dense forest conditions. However, because crown width information may be missing for some large trees, we recorded the crown area as a



Fig. 1. The convolutional neural network structure using the crown width and the DSM with four channels created from the LiDAR point cloud of a tree crown as the inputs.

separate feature alongside the DSM  $\times$  4 representation.

To create the 4  $\times$  2D format, we generated one pair of aerial view images and one pair of side profile view images for each segmented crown (Fig. 2). One image in each pair was created from the leaf-on point cloud, and the other was created from the leaf-off point cloud. As with the DSM  $\times$  4 format, the aerial images for a single tree crown covered a square area of 16  $\times$  16 m, with the crown apex located in the center of the images. The pixel width, however, was set to 25 cm because depth information was not intended to be captured in the aerial view. To create the aerial images, like the DSM  $\times$  4 format, we recorded the intensity of the highest LiDAR point in each pixel. The side profile images were created from vertical profiles of the point clouds, which had a thickness of 75 cm and passed through the crown apex. Each of the side view images captured a square area of  $16 \times 16$  m with a pixel width of 25 cm. The LiDAR point representing the apex was in the top center pixel. We recorded the mean intensity of leaf-on/leaf-off LiDAR points in the profile for each pixel. Although the majority of trees in our dataset are taller than 16 m, most airborne LiDAR points are recorded in the upper parts of the tree crowns and therefore, a 16 m side view height was deemed sufficient to capture the crown structure



Fig. 2. The convolutional neural network structure using the crown width and the tree height along with the four grayscale images created from the LiDAR point cloud of a tree crown as the inputs.

that is represented by the LiDAR points. However, because tree height information was missing from both the aerial and side views, we recorded height and crown width as two separate features alongside the 4x2D representation.

The DSM × 4 format resembles the 3D point cloud data by losing a minimal 3D structure while the 4 × 2D format only captures the 3D data from two 2D views taking the advantage of the symmetry of an ideally shaped tree crown. To augment the data and increase the training data size for deep learning experiments, we created the DSM × 4 and the 4 × 2D representations over 180 rotational variations of each point cloud. We iteratively rotated the point cloud along a nadir axis through the apex by 2° and created a DSM × 4 and a 4 × 2D representation in each iteration. Although the 4 × 2D format loses much of the 3D information because a real tree crown have several dissymmetrical structural features, this information is re-gained when using 180 rotational augmentations per instance.

#### 2.3. Convolutional neural network models

For the DSM  $\times$  4 input format, we stacked six pairs of convolutional and max pooling layers including rectified linear units (ReLUs) as the activation units (Fig. 1). A convolutional layer performs convolution on sliding windows over its input according to the parameters that are trained while a max pooling layer downsizes its input by outputting only the maximum value of windows over the input. Each convolutional layer here included four windows of  $3 \times 3 \times 4$  with a sliding step of one pixel that was operating on a zero-padded input to maintain the same size for the output. Each max pooling layer included  $2 \times 2$  max pooling windows per channel, downsampling the output of the preceding convolutional layer to half of the width and the height. Operating on the representation input of size  $128 \times 128 \times 4$ , this layer composition produces a  $2 \times 2 \times 4$  output structure, which is flattened to 16 output units. On the other hand, for the crown area input feature, we stacked two dense layers, each including two ReLU units. We then put the 16 units initiated from the DSM image and the two units initiated from the crown area feature together and stacked two dense layers of 25 and 10 ReLU units respectively to the end. Finally, we added a softmax layer to obtain the probability distribution over onehot-encoded class labels.

For the 4 × 2D input format, we stacked five pairs of convolutional and max pooling layers including ReLU activation units per each singlechanneled 2D image (Fig. 2). Each convolutional layer included one window of size 3 × 3 with a sliding step of one pixel that was operating on a zero-padded input. Each max pooling layer included windows of  $2 \times 2$ , downsampling the output of the preceding convolutional layer to half of the width and the height. Operating on the set of four image representation inputs of size 64 × 64, this layer compositions produce a  $4 \times 2 \times 2$  output structure, which is flattened to 16 output units. On the other hand, for the crown width and the tree height input features, we stacked two dense layers, including four and two ReLU units, respectively. Like the DSM network, we put the previous 18 units together and added two dense layers of 25 and 10 ReLU units and a final softmax layer respectively to the end.

The DSM × 4 format allows the deep network architecture to perform an early fusion of the leaf-on and leaf-off data as well as the intensity and height values associated with the data. The network captures the correlation between the four channels for the classification task by including more parameters and intermediate features. On the other hand, the 4 × 2D format allows a late fusion to the network, i.e., the leaf-off and leaf-on data and their intensity/height values are not fused until after the corresponding convolutional and max pooling layers produced features independently. While the DSM × 4 format allows for a richer training model, the 4 × 2D format incurs less computational cost.

#### 2.4. Mislabel correction via iterative resampling

As described earlier, registration of the segmented tree crowns to the field-surveyed tree stem locations was done through a probabilistic scoring process. Moreover, the GPS error for the field-surveyed plot centers (~5 m) can exceed the distances between individual trees. These issues likely resulted in a fraction of mis-registrations hence yielding mislabels for the classification task in this work. Mislabeling occurs when a field-surveyed coniferous tree stem is assigned to a segmented deciduous tree crown or vice versa. In the semi-supervised learning literature, a number of studies trained learning models that are robust to such noise by modifying the learning model to explicitly account for the noise (Mnih and Hinton, 2012; Natarajan et al., 2013; Reed et al., 2014), although these studies did not necessarily correct mislabels for external use. Other studies attempted to eliminate/correct mislabels by training learning models and identified mislabels by performing statistical inference on the classification result of the trained models (Bhadra and Hein, 2015; Brodley and Friedl, 1999). These studies either used a small noise-free dataset or, when that was not possible, made assumptions about the tolerable amount of noise in their data to train their learning models for identifying mislabels. For the latter scenario, some studies reported successful identification of mislabels in the presence of up to 40% noise in the training data (Brodley and Friedl, 1999). Unlike general RGB images that are specifically designed for human visual comprehension, remotely sensed LiDAR-represented tree crowns are difficult and uncertain for human experts to classify, making it infeasible to create a noise-free dataset. Therefore, we performed mislabel correction through ensemble filtering (Brodley and Friedl, 1996), which is derived by a series of resampling and statistical inferences.

We built 100 4  $\times$  2D-input networks, and each network was trained using a balanced, random sample of 80 deciduous and 80 conifer instances from our labeled dataset. Random sampling was performed without replacement: once all corresponding labeled instances were used, we started over and continued until all 100 networks were built. This randomization pattern ensured that all instances of a class had (almost) equal contributions across all networks in the training process. Training using a balanced sample for each network was to minimize the effect of the unbalanced training data while having several networks was to take advantage of the entire dataset. To train the networks, we used the Keras deep learning library: we set the loss function to categorical cross entropy and ran the Adam optimizer (learning rate = 0.01) (Kingma and Ba, 2014). Training of each network was performed for three epochs in order to ensure that the process converged to a reasonable state, i.e., the training accuracy was lifted from the base accuracy of 50% but did not reach an overfitting phase.

For each network *n*, we computed the average of the test accuracies of *n* over the 180 augmented forms  $(acc_{ni})$  for every instance *i* in the labeled dataset if i was not used in training n. Assuming instance i is correctly labeled, its test accuracy should on average be equal to the training accuracy of the trained network n ( $acc_n$ ). On the other hand, when instance i is mislabeled, its test accuracy should on average be equal to the symmetric value of the training accuracy of n about the base accuracy of 50% (1 -  $acc_n$ ). Therefore, if  $acc_{ni}$  is less than the symmetric value of the training accuracy of n about 50%, i.e.,  $acc_{ni}$  less than 1 - acc<sub>n</sub>, it is very likely that *i* is mislabeled. Using all 100 networks, we generated values of  $acc_{ni}$  - (1-acc<sub>n</sub>) per each instance i and used these values to perform a T-test on whether their mean was less than zero. If the T-test indicated that an instance was mislabeled, we flipped the label for that instance. We repeated the process of training 100 networks, performing T-tests, and flipping mislabels until no mislabels were identified. Since 2,528 T-tests were performed in each iteration, we used the significance level of  $10^{-8}$  for the T-tests. This significance level, according to the conservative Bonferroni principle, would not allow a false flip rate of more than  $2.5 \times 10^{-5}$  per iteration.



Fig. 3. (a) Rates of flip for coniferous and deciduous trees over the 13 iterations of the mislable correction process; and (b) average training accuracy of 100 networks over the 13 iterations.

## 2.5. Classification and evaluation

After correcting potential mislabels, we used an ensemble of 50 networks to perform the classification. We trained each network on a balanced, random sample of 100 deciduous and 100 coniferous instances using the Adam optimizer with a learning rate of 0.01. Like the mislabel correction procedure, random sampling was performed without replacement. To produce cross-validated classification accuracies, for each instance in the dataset, we performed classification according to the average softmax probabilities produced only by the networks that did not use that instance for training, and validated the classification against the instance label. We performed the same ensemble cross validation procedure for both the DSM  $\times$  4 and the  $4 \times 2D$  formats. To compare early and late fusion, we performed two additional experiments. We designed a network for four separate singlechanneled DSMs that included six pairs of convolutional and max pooling layers (one 3 × 3 filter per convolutional layer, ReLU activation units, and  $2 \times 2$  pooling windows). We also designed another network for two double-channeled 2D images (aerial view images were used for the two channels of one image and side profile views were used for the two channels of the other image) that included five pairs of convolutional and max pooling layers (two filters of  $3 \times 3 \times 2$  per convolutional layer, ReLU units, and  $2 \times 2$  pooling windows). For both of these networks the rest of the layers were identical to the corresponding 2D/DSM network designs presented in Section 2.3(Figs. 1 and 2). The training was run for fifteen epochs for every DSM-based network, but five epochs appeared to be enough for every 2D image-based network.

To compare the performance of the proposed deep CNN learning models with the shallow learning methods used in the previous work, we assembled five features that we believe capture sufficient information required for the classification task. For each instance, we retrieved tree height, crown width, mean intensity of leaf-off points, mean normalized intensity of leaf-on points, and proportion of leaf-on points to the leaf-off ones. We scaled the values of these features to be between zero and one across the dataset. We used logistic regression, KNN, SVMs, LDA, quadradic discriminant analysis (QDA), random forest, and multi-layer perceptron (two hidden layers of eight and four ReLUs respectively) as the methods to run our shallow learning experiments. For each of the shallow learning methods, we performed the same ensemble experiment as we did for the deep learning models to produce comparable cross-validated accuracies.

For further experiments on deep learning for conifer/deciduous classification of LiDAR-represented tree crowns, we used the  $4 \times 2D$  format because of the lower computational load. To investigate the effect of the training data size, we created stratified random subsamples of our dataset. We subsampled 20%, 40%, ..., 100% of the deciduous

and coniferous trees and performed the cross validated classification procedure described above for each subsampled dataset. We adjusted the size of resampling instances in proportion to the subsample size, though the number of ensemble networks was held constant. To quantify the effect of data augmentation, we measured the accuracies for when 20, 40, ..., 180, 240, 300, and 360 rotations of each instance were included. We then looked into the effects of the domain parameters: we ran the cross validation experiment excluding leaf-off data, excluding leaf-on data, using non-normalized intensities for leaf-on data, excluding intensity values (using binary values representing existence of a point per pixel) and excluding height and crown width features. When excluding leaf-on and leaf-off data, we decreased the size of the last two dense layers before the softmax layer to 16 and 8 units respectively to account for the smaller input size. We also inspected the correlation between the point density of a crown cloud and the probability of the softmax output unit associated with the correct label of the crown cloud to determine how point density affected the classification accuracy. Lastly, we stratified the classification result to overstory (dominant and co-dominant) and understory (intermediate and overtopped) trees to inspect how crown class affected the classification performance.

## 3. Results and discussion

## 3.1. Mislabel correction

The process of mislabel correction converged after 13 iterations and increased the number of conifers from 124 to 214 and decreased the number of deciduous trees from 2404 to 2314 (Fig. 3-a). According to the original field measurements (Table 1), 7.27% of the trees in RF are conifers, which is slightly lower than the result after correcting mislabels -8.46% conifers. The reason for this slight difference may be the relative difficulty in segmenting deciduous trees compared to coniferous trees due to the variety of crown shapes and the looser, interwoven foliage, which creates complicated, difficult-to-distinguish LiDAR point patterns (Vauhkonen et al., 2012). This effect likely resulted in larger rate of undetected deciduous trees after segmentation and registration with the field data. In total, the labels for 35 of the initial 124 (28.22%) conifers and 125 of 2404 (5.20%) initial deciduous trees were flipped. These unbalanced flip rates concur with the dominant presence of deciduous trees, i.e., if a field deciduous tree is misregistered to a LiDAR crown, the crown is likely another deciduous tree (yielding no mislabel) while this is not the case for a mis-registered field conifer. Over the 13 iterations of the mislabel correction procedure, the average training accuracy of the 100 networks started at 67.1% and plateaued at 83.6% (Fig. 3-b). This trend suggests that a number of highly likely (controlled by the T-tests) mislabels were corrected,



Fig. 4. Classification accuracy for different shallow and deep learning methods.

improving the model accuracy, while less likely mislabels were left unchanged, resulting in the accuracy plateau and prohibition of overfitting. Overall, the mislabel correction process produced more realistic labels by increasing the number of coniferous trees from 4.90% to 8.46% within the 2528 segmented tree crowns.

## 3.2. Classification accuracy

The cross-validated accuracies associated with the shallow learning methods for conifers ranged from 78.5  $\pm$  5.5% (random forest) to 81.3  $\pm$  5.2% (KNN) and for deciduous trees ranged from 75.8  $\pm$  1.8% (QDA) to 87.4  $\pm$  1.4% (random forest) at a 95% confidence level (Fig. 4). While the accuracies for conifers were not significantly different, the accuracies for deciduous trees showed significant differences across the shallow learning methods. Logistic regression, LDA, and QDA showed relatively lower accuracies, suggesting that the strong biases in their internal modeling structure compared to the rest of the methods caused them not to fit to the data as effectively.

For the deep learning methods, the accuracies associated with the DSM  $\times$  4 representation were 80.4  $\pm$  5.3% for conifers and 90.1  $\pm$  1.3% for deciduous trees. The equivalent classification accuracies associated with the 4  $\times$  2D representation were 82.7  $\pm$  5.1% and 90.2  $\pm$  1.3%, respectively for coniferous and deciduous trees (Fig. 4). The slight, insignificant higher accuracies associated with the 4  $\times$  2D representation is likely due to the fact that we used this format for the mislabel correction process, which might have negligibly biased the data. The experiments with the four single-channeled DSMs and two double-channeled 2D images formats showed insignificant differences in the accuracies. This observation indicates that the designed representations include sufficient information from the raw data such that any reasonable network architecture may draw the required information for classification.

As mentioned, the DSM  $\times$  4 format more closely resembles 3D data, which together with the richer early-fused network, have the potential to achieve higher classification accuracies. However, the 4  $\times$  2D format with a late-fused network could achieve similar accuracies.

The deep learning methods showed slightly better classification accuracies for conifers (by up to 5%), although they were not statistically significant. The reason is likely the small size of the original conifer sample, rendering insufficient statistical power. However, for the deciduous trees, the deep learning methods showed statistically significant better accuracies (by 3–14%) (Fig. 4). As the deep learning methods process the entire information rather than a set of manually designed features, they can automatically and objectively derive the useful information and hence provide the ground for better classification.

Samples of correct and incorrect classifications using the 4x2D format for understory and overstory trees are visualized in Fig. 5 and Fig. 6. As shown, no clear pattern to distinguish conifer from deciduous is present.

## 3.3. Effect of training data size on the classification

Increasing the size of training data improved the classification accuracies. For deciduous trees the accuracy plateaued when using only 40% of the original dataset (~925 deciduous and ~86 coniferous trees) but for coniferous trees, the accuracy did not plateau even when all of the 214 training instances were included (Fig. 7). This observation suggests that accuracy can be increased simply by collecting more conifer training instances.

## 3.4. Effect of data augmentation on classification

Including a greater number of rotational augmentations per instance



Fig. 5. Sample visualization of correct and incorrect classifications for overstory trees.

slightly improved the classification accuracies. Using only 20 rotations per instance resulted in 73.8% accuracy for coniferous trees and 87.7% accuracy for deciduous trees, which are lower than when using the original 180 rotations. The improvement in classification plateaued at ~60 rotations for deciduous trees and ~150 rotations for coniferous trees (Fig. 8). Having more deciduous trees likely resulted in a smaller number of rotations/augmentations to be enough for the classification task. Although a higher number of rotations could compensate for the small number of coniferous training instances to some extent, augmentations are unlikely to match the classification quality provided by a higher number of real training instances.

## 3.5. Effect of domain data on classification

Excluding the leaf-off data resulted in a remarkable decrease in classification accuracy for the conifers (from 82.7% to 61.2%) and a small decrease in accuracy for deciduous trees (from 90.2% to 89.6%), while excluding the leaf-on data resulted in a minor decrease in accuracy for conifers (from 82.7% to 81.6%) and a negligible increase in accuracy for deciduous trees (from 90.2% to 90.5%) (Fig. 5). This observation indicates that, despite the much lower point density, the leaf-off data provided the most useful features for the classification task, which concurs with the result of the previous work (Kim et al., 2011; Reitberger et al., 2008). As conifers have perennial foliage, the leaf-off LiDAR points could represent their crown shapes even at a low density while the deciduous trees may only be represented by a few random LiDAR points returning from their defoliated branches. The dense leaf-

on data could on the other hand represent the crown shapes for both conifers and deciduous trees and was used here for segmentation of the individual tree crowns. Attempting to distinguish the crown shapes of deciduous and coniferous trees as provided by the leaf-on data is likely less efficient than distinguishing between a random point pattern (a deciduous tree) and a crown-like shape (a coniferous tree) as provided by the leaf-off data. However, for identifying species, which is a more complicated classification task and a subject of future work, the highdensity leaf-on data may be more useful.

Using binary values instead of the intensity values resulted in a decrease in classification accuracy for conifers (from 82.7% to 69.2%) and only a negligible increase in accuracy for deciduous trees (from 90.2% to 91.1%) (Fig. 9). Using the normalized intensity values for the leaf-on data (when excluding the leaf-off data) compared with using non-normalized values seemed to make minor, insignificant improvements in the classification accuracies for conifers (from 60.3% to 61.2%) and deciduous trees (from 88.9% to 89.6%) (Fig. 9). Although LiDAR intensity values were useful for the classification, normalizing the intensity values yielded no significant improvement. Excluding the tree height and crown width features yielded slight, insignificant increase in classification accuracy of conifers (from 82.7% to 83.4%) and a slightly stronger decrease in the accuracy of deciduous trees (from 90.2% to 88.3%) (Fig. 9). This observation indicates that the crown dimension features did not have a strong effect in distinguishing between coniferous and deciduous trees. The caveats however are that (i) the crown width was partially captured in the aerial views of the 4x2D format; and (ii) the closedness of the forest canopy resulted in smaller crown footprints of the trees to be captured via



Fig. 6. Sample visualization of correct and incorrect classifications for understory trees.



**Fig. 7.** Classification accuracies measured against the size of the training data. Each symbol in the diagram represents the average of 20 observations.

LiDAR irrespective of the tree type. Bearing these caveats, a similar deep learning method is still likely effective for area/patch-based (as opposed to individual crown-based) conifer/deciduous classification task in natural, closed-canopy forests.

Some of the domain data such as the leaf-off data and the intensity values appeared to be important in the classification task, which is evident in the changes in the accuracy for conifers (Fig. 9). However, as can be observed in only slight changes in the accuracy of deciduous trees, abundance of training data likely compensated for the absence of a subset of the important domain data.

## 3.6. Effects of crown class and point density on classification

For overstory trees, the cross validated classification accuracy was 92.1  $\pm$  4.7% for conifers and 87.2  $\pm$  2.2% for deciduous trees. The classification accuracy for understory trees was 69.0  $\pm$  9.8% for conifers and 92.1  $\pm$  1.4% for deciduous trees (Fig. 10). The crown of an understory tree is typically captured only partially by airborne LiDAR (Fig. 6), as it is covered by the overstory trees. The partial shapes of these crowns decrease the classification power, likely yielding the correlated accuracies to become easily biased by the abundance of deciduous instances compared with coniferous instances. In contrast, the crowns of overstory trees are captured more completely (Fig. 5) allowing for a more powerful classification. Lastly, we could not identify any significant correlation between point density (neither leaf-off nor leaf-on) and the classification accuracy (neither for overstory nor for understory trees). This observation does not concur with previous work reporting a positive correlation between accuracy and point density (Li et al., 2013). The reason is likely that the classification task is primarily driven by the leaf-off data, the point density range of which is too small (0.1–6.0  $pt/m^2$  for the middle 95%) to surface any effect. Moreover, the partial crowns captured may feature high point densities but are not easy to classify due to their incomplete shapes (Fig. 6).



Fig. 8. Classification accuracies measured against the number of rotational augmentations per instance.



Fig. 10. Classification accuracy of overstory and understory trees.

#### 4. Conclusions

Airborne LiDAR point clouds representing individual trees can be used to predict tree attributes such as tree type. Previous work exploited shallow learning techniques that require the engineering of useful features by a human expert. In this work, we eliminated the need for feature engineering by a human expert by using deep learning CNNs to classify crown point clouds as coniferous or deciduous trees. We segmented individual trees from the LiDAR point clouds and registered them with field-surveyed trees to create training data. We designed two different discrete representations of a crown's 3D point cloud to enable its processing by a deep CNN. We benchmarked accuracy of the deep learning against multiple shallow learning methods. We also investigated the effect of training data preparation, CNN design, as well as domain data on the accuracy of the classification.

In addition to automatic derivation of the features, the deep CNN learning methods showed improved classification accuracies compared with shallow learning methods. Our investigation of the coniferous/ deciduous deep learning classification showed that a set of 2D views/

profiles of a 3D point cloud are not only more efficient to be processed but also can yield similar accuracies compared with bulkier 2.5D (or even 3D) representations. The results presented indicate that deep learning can effectively and efficiently be used for classifying tree type based on airborne LiDAR point clouds representing individual tree crowns, which is a step forward to operational tree-level remote quantification of large-scale forests. Although further experiments using richer datasets and for more complicated prediction tasks (e.g., species classification) are required, deep learning provides the feasibility of automatic extraction of optimal features toward the prediction task. This unique deep learning characteristic brings about the potentials for successful prediction tasks in different domains such as remote sensing and biomedical image analysis, where the data modalities are not friendly to the human perceptual system and, given the large amount of information contained in the data, have likely operated using suboptimal human-designed features.



Fig. 9. Classification accuracy when excluding domain data.

#### **Declaration of Competing Interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Acknowledgments

This work was supported by: (1) the Department of Forestry at the University of Kentucky and the McIntire-Stennis project KY009026 Accession 1001477, (ii) the Kentucky Science and Engineering Foundation under the grant KSEF-3405-RDE-018, and (iii) the University of Kentucky Centre for Computational Sciences.

## Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.isprsjprs.2019.10.011.

#### References

- Ackermann, F., 1999. Airborne laser scanning—present status and future expectations. ISPRS J. Photogramm. Remote Sens. 54, 64–67.
- Allen, M.P., 1997. Partial regression and residualized variables. Understan. Regression Anal. 86–90.
- Amiri, N., Yao, W., Heurich, M., Krzystek, P., Skidmore, A.K., 2016. Estimation of regeneration coverage in a temperate forest by 3D segmentation using airborne laser scanning data. Int. J. Appl. Earth Obs. Geoinf. 52, 252–262.
- Bergstra, J., Bengio, Y., 2012. Random search for hyper-parameter optimization. J. Mach. Learn. Res. 13, 281–305.
- Bhadra, S., Hein, M., 2015. Correction of noisy labels via mutual consistency check. Neurocomputing 160, 34–52.
- Blomley, R., Hovi, A., Weinmann, M., Hinz, S., Korpela, I., Jutzi, B., 2017. Tree species classification using within crown localization of waveform LiDAR attributes. ISPRS J. Photogramm. Remote Sens. 133, 142–156.
- Brodley, C.E., Friedl, M.A., 1996. Improving automated land cover mapping by identifying and eliminating mislabeled observations from training data. In: Geoscience and Remote Sensing Symposium, 1996. IGARSS'96. Remote Sensing for a Sustainable Future.', International, IEEE, pp. 1379–1381.
- Brodley, C.E., Friedl, M.A., 1999. Identifying mislabeled training data. J. Artif. Intell. Res. 11, 131–167.
- Bruggisser, M., Roncat, A., Schaepman, M.E., Morsdorf, F., 2017. Retrieval of higher order statistical moments from full-waveform LiDAR data for tree species classification. Remote Sens. Environ. 196, 28–41.
- Cao, L., Coops, N.C., Innes, J.L., Dai, J., Ruan, H., She, G., 2016. Tree species classification in subtropical forests using small-footprint full-waveform LiDAR data. Int. J. Appl. Earth Obs. Geoinf. 49, 39–51.
- Carpenter, S.B., Rumsey, R.L., 1976. Trees and shrubs of Robinson Forest Breathitt County, Kentucky. Castanea 277–282.
- Department of Forestry, 2007. Robinson Forest: a facility for research, teaching, and extension education. In: University of Kentucky.
- Dou, Q., Chen, H., Yu, L., Zhao, L., Qin, J., Wang, D., Mok, V.C., Shi, L., Heng, P.-A., 2016. Automatic detection of cerebral microbleeds from MR images via 3D convolutional neural networks. IEEE Trans. Med. Imaging 35, 1182–1195.
- Duncanson, L., Dubayah, R., Cook, B., Rosette, J., Parker, G., 2015. The importance of spatial detail: assessing the utility of individual crown information and scaling approaches for lidar-based biomass density estimation. Remote Sens. Environ. 168, 102–112.
- Evans, J.S., Hudak, A.T., Faux, R., Smith, A., 2009. Discrete return lidar in natural resources: recommendations for project planning, data processing, and deliverables. Remote Sens. 1, 776–794.
- Farfade, S.S., Saberian, M.J., Li, L.-J., 2015. In: Multi-view Face Detection Using Deep Convolutional Neural Networks. ACM, Shanghai, China, pp. 643–650.
- Franklin, S.E., 2001. Remote Sensing for Sustainable Forest Management. CRC Press. Gatziolis, D., 2011. Dynamic range-based intensity normalization for airborne, discrete
- return lidar data of forest canopies. Photogramm. Eng. Remote Sens. 77, 251–259. Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference
- on Computer Vision and Pattern Recognition, pp. 580–587. Gougeon, F.A., 1995. A crown-following approach to the automatic delineation of in-
- dividual tree crowns in high spatial resolution aerial images. Can. J. Remote Sens. 21, 274–284.
- Guan, H., Yu, Y., Ji, Z., Li, J., Zhang, Q., 2015. Deep learning-based tree classification using mobile LiDAR data. Remote Sens. Lett. 6, 864–873.
- Hamraz, H., Contreras, M.A., Zhang, J., 2016. A robust approach for tree segmentation in deciduous forests using small-footprint airborne LiDAR data. Int. J. Appl. Earth Obs. Geoinf. 52, 532–541.
- Hamraz, H., Contreras, M.A., Zhang, J., 2017a. Forest understory trees can be segmented accurately within sufficiently dense airborne laser scanning point clouds. Sci. Rep. 7,

#### 6770.

- Hamraz, H., Contreras, M.A., Zhang, J., 2017b. A scalable approach for tree segmentation within small-footprint airborne LiDAR data. Comput. Geosci. 102, 139–147.
- Hamraz, H., Contreras, M.A., Zhang, J., 2017c. Vertical stratification of forest canopy for segmentation of understory trees within small-footprint airborne LiDAR point clouds. ISPRS J. Photogramm. Remote Sens. 130, 385–392.
- Harikumar, A., Bovolo, F., Bruzzone, L., 2017. An internal crown geometric model for conifer species classification with high-density LiDAR data. IEEE Trans. Geosci. Remote Sens. 55, 2924–2940.
- Holmgren, J., Persson, Å., 2004. Identifying species of individual trees using airborne laser scanner. Remote Sens. Environ. 90, 415–423.
- Holopainen, M., Talvitie, M., 2007. Effect of data acquisition accuracy on timing of stand harvests and expected net present value. Silva Fennica 40, 531.
- Hyyppä, J., Yu, X., Hyyppä, H., Vastaranta, M., Holopainen, M., Kukko, A., Kaartinen, H., Jaakkola, A., Vaaja, M., Koskinen, J., 2012. Advances in forest inventory using airborne laser scanning. Remote Sens. 4, 1190–1207.
- Jing, L., Hu, B., Li, J., Noland, T., 2012. Automated delineation of individual tree crowns from LiDAR data by multi-scale analysis and segmentation. Photogramm. Eng. Remote Sens. 78, 1275–1284.
- Kalogerakis, E., Averkiou, M., Maji, S., Chaudhuri, S., 2017. 3D shape segmentation with projective convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2.
- Kashani, A.G., Olsen, M.J., Parrish, C.E., Wilson, N., 2015. A Review of LiDAR radiometric processing: from Ad Hoc intensity correction to rigorous radiometric calibration. Sensors 15, 28099–28128.
- Kim, S., Hinckley, T., Briggs, D., 2011. Classifying individual tree genera using stepwise cluster analysis based on height and intensity metrics derived from airborne laser scanner data. Remote Sens. Environ. 115, 3329–3342.
- Kingma, D., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Korpela, I., Ørka, H.O., Maltamo, M., Tokola, T., Hyyppä, J., 2010. Tree species classification using airborne LiDAR–effects of stand and tree parameters, downsizing of training set, intensity normalization, and sensor type. Silva Fennica 44, 319–339.
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.
- Kuhn, H.W., 1955. The Hungarian method for the assignment problem. Naval Res. Logist. Quarterly 2, 83–97.
- Kwak, D.-A., Lee, W.-K., Lee, J.-H., Biging, G.S., Gong, P., 2007. Detection of individual trees and estimation of tree height using LiDAR data. J. Forest Res. 12, 425–434.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436-444.

- Leica Geosystems, 2013. Leica ALS60, Airborne Laser Scanner Product Specifications. In: Li, J., Hu, B., Noland, T.L. (Eds.), Classification of Tree Species based on Structural Features Derived from High Density LiDAR Data. Agric. Forest Meteorol. 171, pp. 104–114.
- Li, J., Hu, B., Noland, T.L., 2013. Classification of tree species based on structural features derived from high density LiDAR data. Agric. For. Meteorol. 171, 104–114.
- Lin, Y., Hyyppä, J., 2016. A comprehensive but efficient framework of proposing and validating feature parameters from airborne LiDAR data for tree species classification. Int. J. Appl. Earth Obs. Geoinf. 46, 45–55.
- Lindberg, E., Eysn, L., Hollaus, M., Holmgren, J., Pfeifer, N., 2014. Delineation of tree crowns and tree species classification from full-waveform airborne laser scanning data using 3-D ellipsoidal clustering. IEEE J. Select. Top. Appl. Earth Observat. Remote Sens. 7, 3174–3181.
- Maltamo, M., Næsset, E., Vauhkonen, J., 2014. Forestry Applications of Airborne Laser Scanning: Concepts and Case Studies. Manag For Ecosys.
- Maturana, D., Scherer, S., 2015. Voxnet: A 3d convolutional neural network for real-time object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 922–928.
- Mizoguchi, T., Ishii, A., Nakamura, H., Inoue, T., Takamatsu, H., 2017. Lidar-based individual tree species classification using convolutional neural network. In: Videometrics, Range Imaging, and Applications XIV, International Society for Optics and Photonics, pp. 1033200.
- Mnih, V., Hinton, G.E., 2012. Learning to label aerial images from noisy data. In: Proceedings of the 29th International Conference on Machine Learning (ICML-12), pp. 567–574.
- Natarajan, N., Dhillon, I.S., Ravikumar, P.K., Tewari, A., 2013. Learning with noisy labels. In: Advances in Neural Information Processing Systems, pp. 1196–1204.
- Ørka, H.O., Næsset, E., Bollandsås, O.M., 2009. Classifying species of individual trees by intensity and structure features derived from airborne laser scanner data. Remote Sens. Environ. 113, 1163–1174.
- Overstreet, J., 1984. Robinson Forest inventory. Department of Forestry, University of Kentucky, Lexington, Kentucky.
- Paris, C., Valduga, D., Bruzzone, L., 2016. A hierarchical approach to three-dimensional segmentation of LiDAR data at single-tree level in a multilayered forest. IEEE Trans. Geosci. Remote Sens. 54, 4190–4203.
- Pitkänen, J., 2001. Individual tree detection in digital aerial images by combining locally adaptive binarization and local maxima methods. Can. J. For. Res. 31, 832–844.
- Popescu, S.C., Zhao, K., 2008. A voxel-based lidar method for estimating crown base height for deciduous and pine trees. Remote Sens. Environ. 112, 767–781.
- Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, vol. 1, pp. 4.
- Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J., 2016. Volumetric and multiview cnns for object classification on 3d data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5648–5656.

- Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, pp. 5099–5108.
- Quackenbush, L.J., Hopkins, P.F., Kinn, G.J., 2000. Developing forestry products from high resolution digital aerial imagery. PE&RS, Photogramm. Eng. Remote Sens. 66, 1337–1346.
- Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., Rabinovich, A., 2014. Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412. 6596.
- Reitberger, J., Krzystek, P., Stilla, U., 2008. Analysis of full waveform LIDAR data for the classification of deciduous and coniferous trees. Int. J. Remote Sens. 29, 1407–1431.
- Riegler, G., Osman Ulusoy, A., Geiger, A., 2017. Octnet: Learning deep 3d representations at high resolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3577–3586.
- Roth, H.R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., Kim, L., Summers, R.M., 2016. Improving computer-aided detection using convolutional neural networks and random view aggregation. IEEE Trans. Med. Imaging 35, 1170–1181.
- Sačkov, I., Hlásny, T., Bucha, T., Juriš, M., 2017. Integration of tree allometry rules to treetops detection and tree crowns delineation using airborne lidar data. iForest-Biogeosci. Forestry 10, 459.
- Schmidhuber, J., 2015. Deep learning in neural networks: An overview. Neural Netw. 61, 85–117.
- Socher, R., Huval, B., Bath, B., Manning, C.D., Ng, A.Y., 2012. Convolutional-recursive deep learning for 3d object classification. In: Advances in Neural Information Processing Systems, pp. 656–664.

- Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-view convolutional neural networks for 3d shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 945–953.
- Terrasolid Ltd., 2012. TerraScan User's Guide. In: Terrasolid Oy.
- Vauhkonen, J., Ene, L., Gupta, S., Heinzel, J., Holmgren, J., Pitkänen, J., Solberg, S., Wang, Y., Weinacker, H., Hauglin, K.M., Lien, V., Packalén, P., Gobakken, T., Koch, B., Næsset, E., Tokola, T., Maltamo, M., 2012. Comparative testing of single-tree detection algorithms under different types of forest. Forestry: Int. J. Forest Res. 85, 27–40.
- Vauhkonen, J., Korpela, I., Maltamo, M., Tokola, T., 2010. Imputation of single-tree attributes using airborne laser scanning-based height, intensity, and alpha shape metrics. Remote Sens. Environ. 114, 1263–1276.
- Véga, C., Hamrouni, A., El Mokhtari, S., Morel, J., Bock, J., Renaud, J.-P., Bouvier, M., Durrieu, S., 2014. PTrees: a point-based approach to forest tree extraction from lidar data. Int. J. Appl. Earth Obs. Geoinf. 33, 98–108.
- Wang, Y., Weinacker, H., Koch, B., 2008. A lidar point cloud based procedure for vertical canopy structure analysis and 3D single tree modelling in forest. Sensors 8, 3938–3951.
- Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3d shapenets: a deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1912–1920.
- Yu, X., Hyyppä, J., Vastaranta, M., Holopainen, M., Viitala, R., 2011. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. ISPRS J. Photogramm. Remote Sens. 66, 28–37.